

10

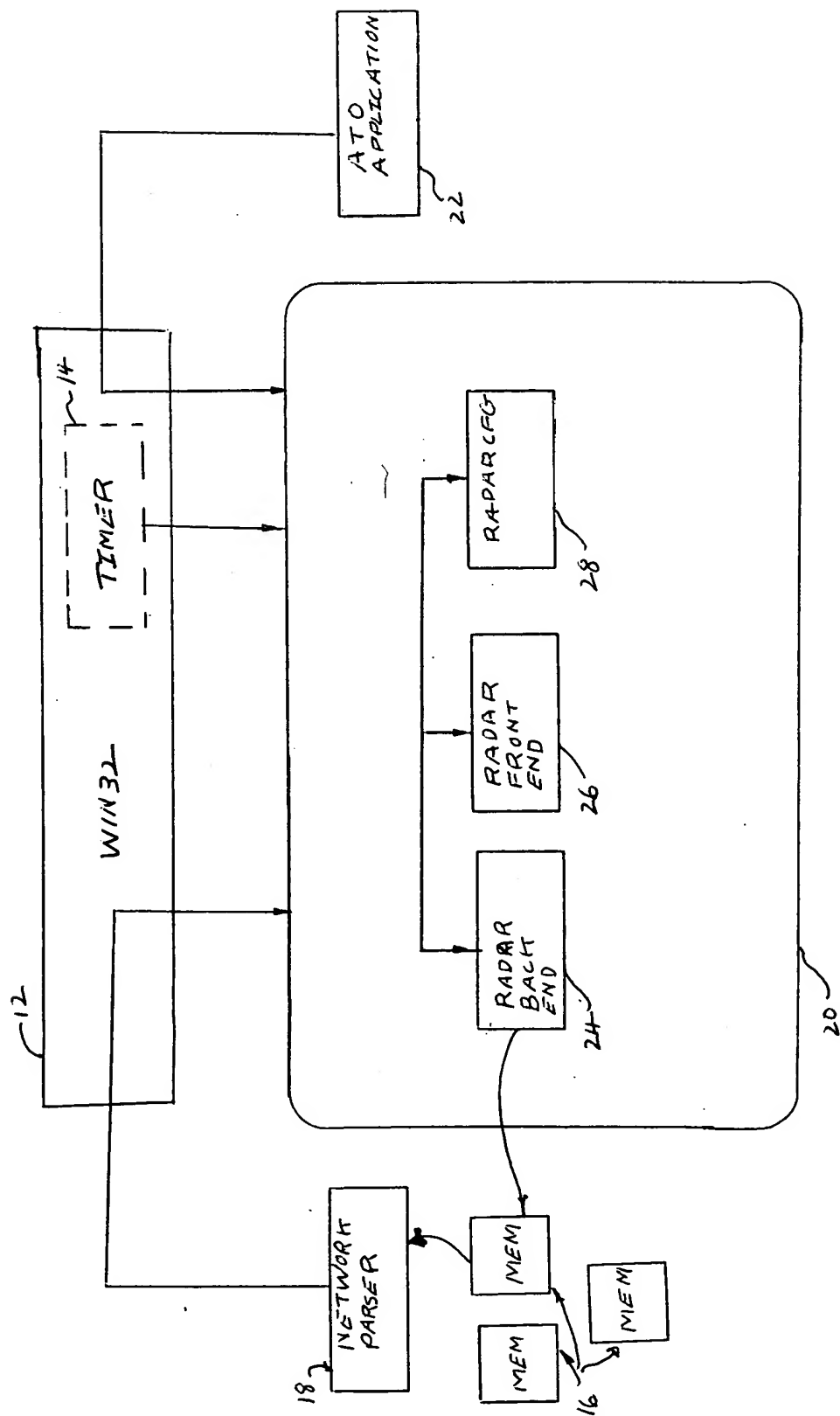
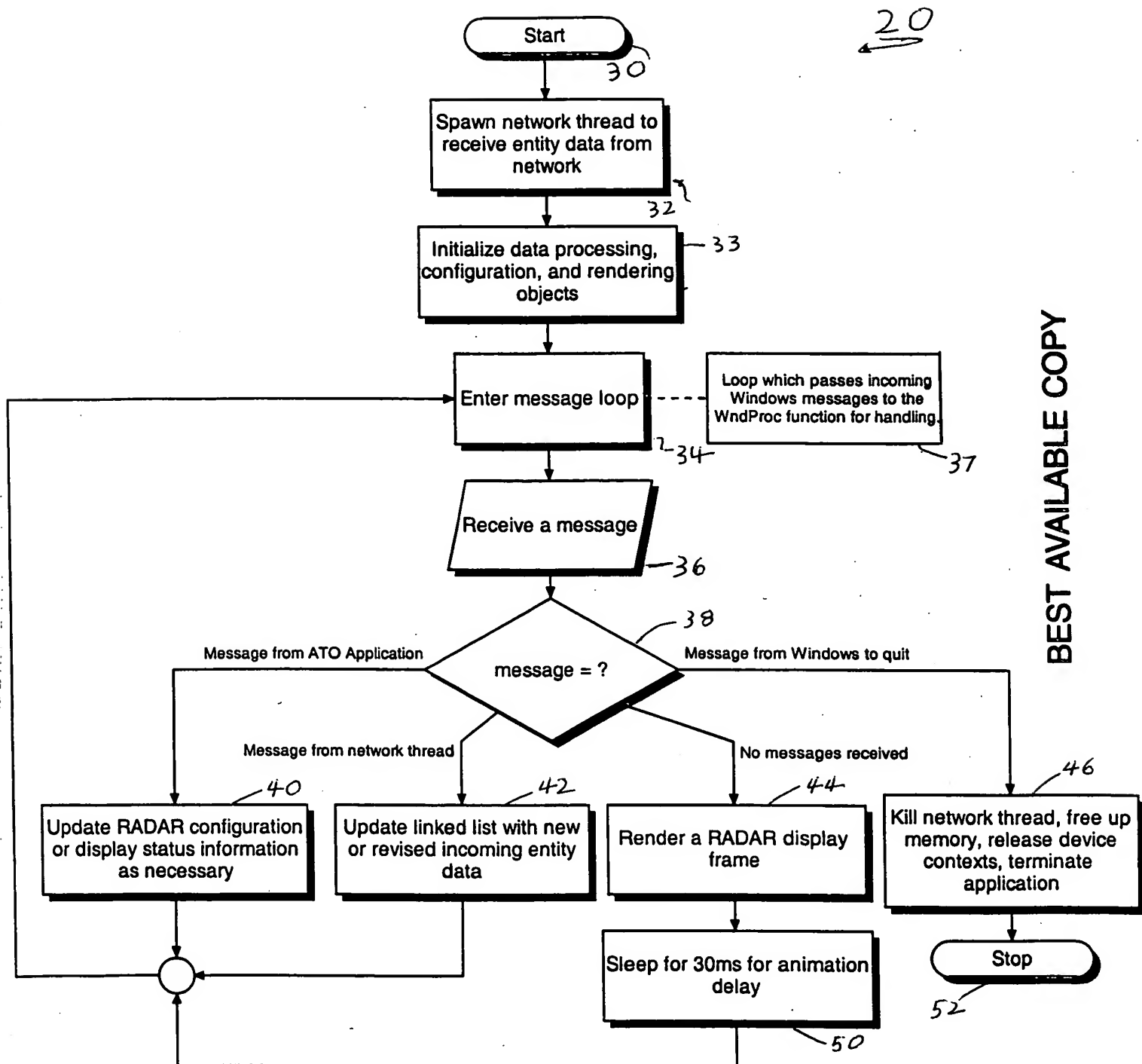


FIG 1

BEST AVAILABLE COPY



BEST AVAILABLE COPY

FIG 2

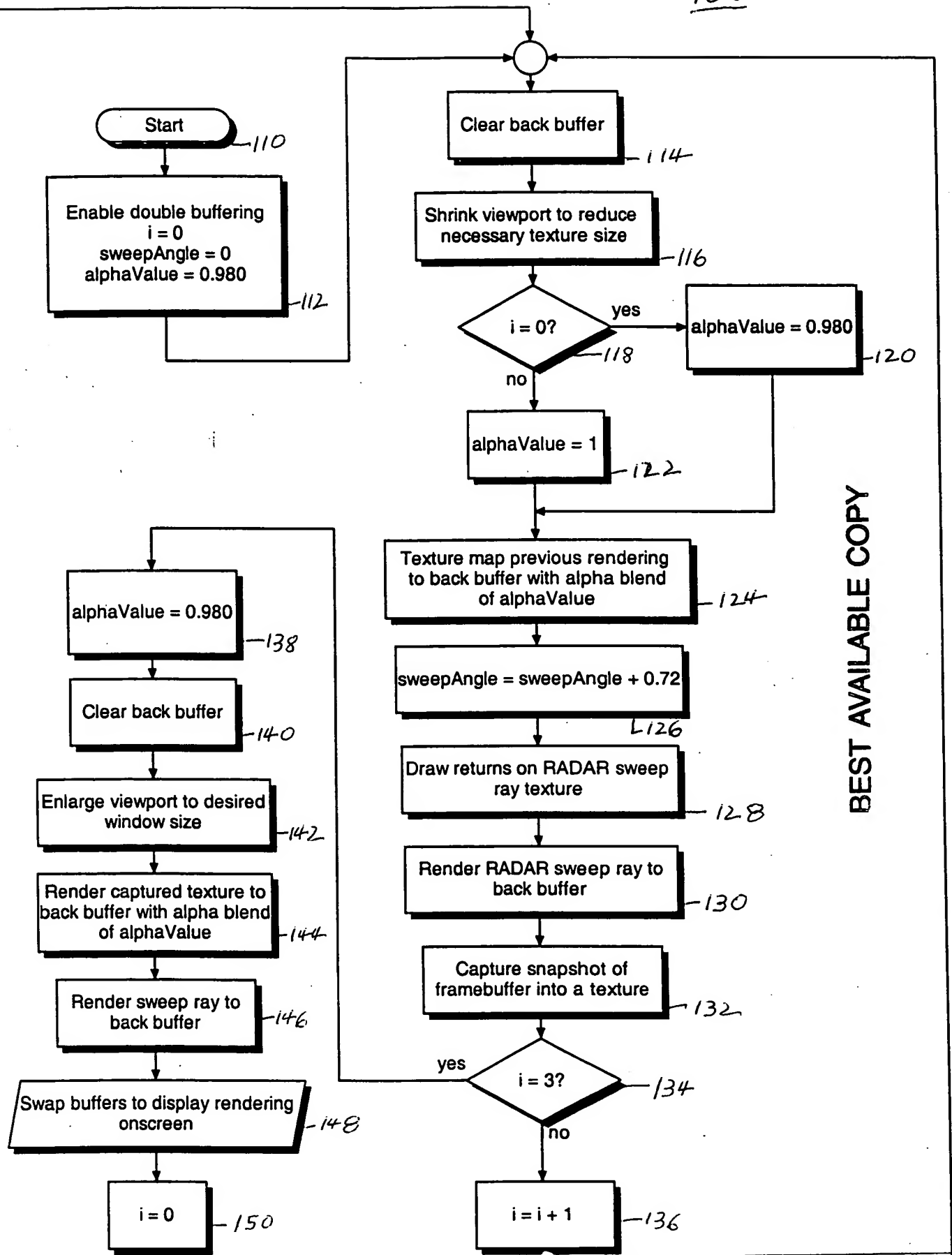


FIG 3

24

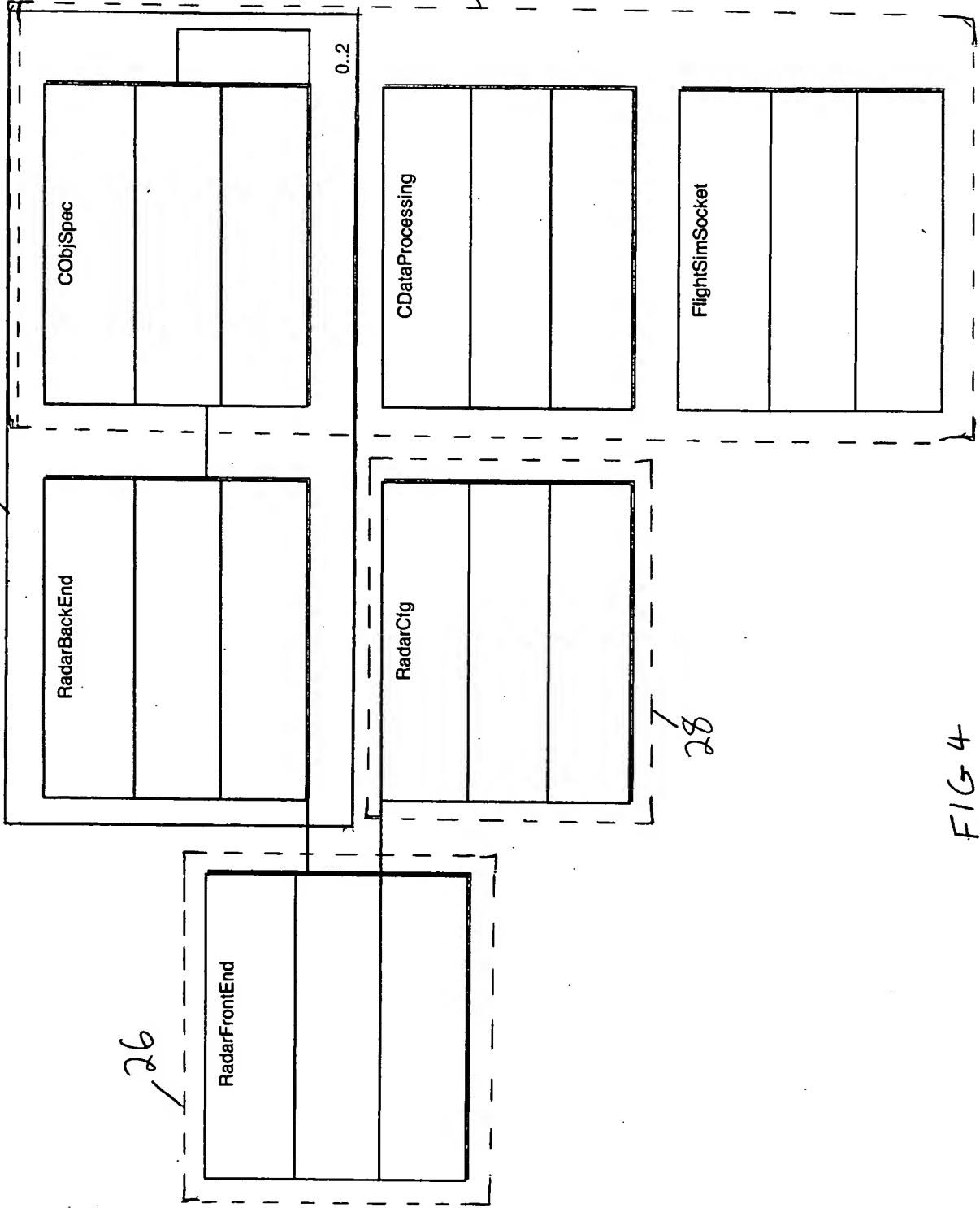


FIG 4

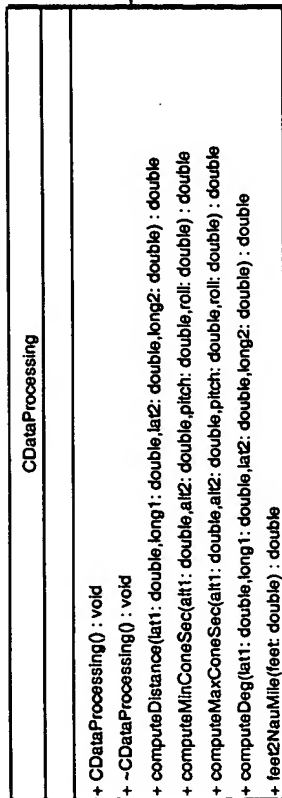
BEST AVAILABLE COPY

26

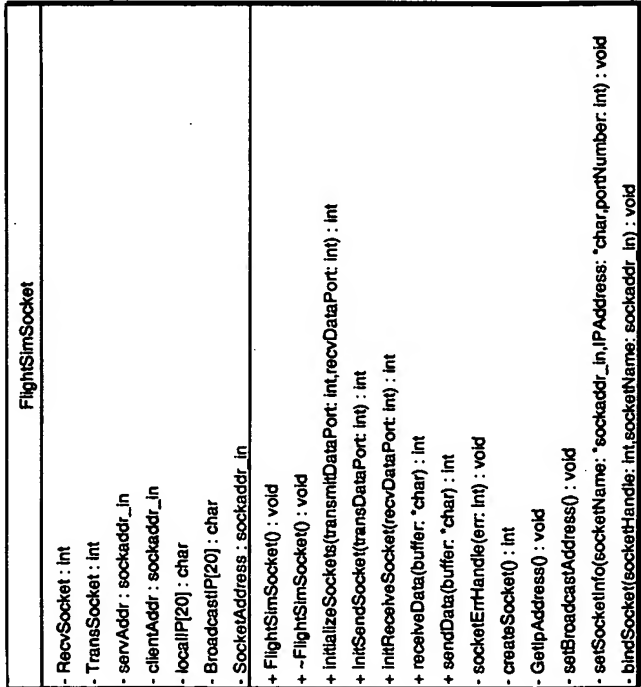
RadarFrontEnd
<ul style="list-style-type: none"> - bRotate : bool - bClearDisp : bool - bStopRendering : bool - bStandby : bool - iLinearSize : GLuint - cxCenter : float - cyCenter : float - fSweepAngle : float - fAlphaFade : float - fSweepIncrement : float - fIRange : float - fGainFactor : float - hpTextures[10] : GLuint - lpSweepTexture[128][4] : GLfloat - upRenderTexture[85538][3] : GLuint - pRadarBackEnd : *RadarBackEnd - pRadarCtg : *RadarCtg - pFirstInt : *ObjSpec
<ul style="list-style-type: none"> + RadarFrontEnd(pConfig: RadarCtg, pBackEnd: RadarBackEnd, cxWidth: GLint, cyHeight: GLint) : void + ~RadarFrontEnd() : void + renderScene() : void + updateParameters() : void + pauseRendering() : void + continueRendering() : void + getHelpYaw() : void - orthoMode(xLeft: GLint, xRight: GLint, yBottom: GLint, yTop: GLint) : void - perspectiveMode() : void - createSweep(uTextureID: GLuint, lxCenter: GLfloat, lyCenter: GLfloat, lxWidth: GLfloat, lxHeight: GLfloat, lyLength: GLfloat, lyHeight: GLfloat) : void - createTexture(uTextureID: GLuint) : void - renderMotionBlur(uTextureID: GLuint) : void - renderHelpSymbol() : void - drawBlip() : void

FIG. 4 BEST AVAILABLE COPY

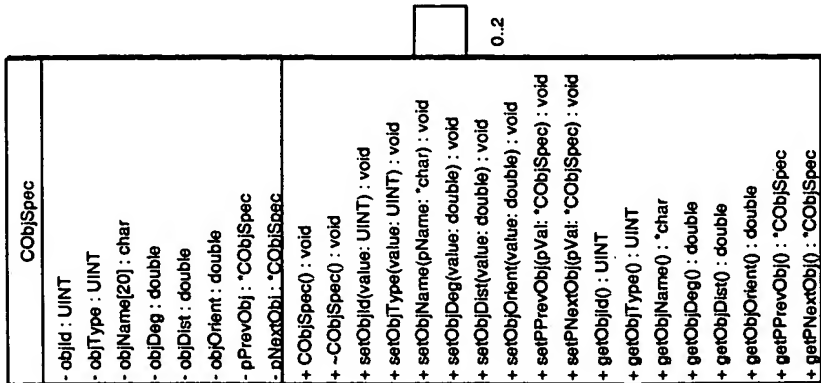
RadarCtg	
- iModeSpeed : UINT	
- iRcvrGain : UINT	
- iStab : UINT	
- iEraseGPI : UINT	
- iPersist : UINT	
- iRange : UINT	
- cxHelo : UINT	
- cyHelo : UINT	
- cxHeloOffset : UINT	
+ RadarCtg(rModeSpeed: UINT, rRcvrGain: UINT, rStab: int, rEraseGPI: int, rPersist: UINT, rRange: UINT, rXPos: UINT, rYPos: UINT, rYOffset: UINT) : void	
+ setModeSpeed(rParam: UINT) : void	
+ setRcvrGain(rParam: UINT) : void	
+ setStab(rParam: UINT) : void	
+ setEraseGPI(rParam: UINT) : void	
+ setPersist(rParam: int) : void	
+ setRange(rParam: UINT) : void	
+ setHeloXPos(rParam: UINT) : void	
+ setHeloYPos(rParam: UINT) : void	
+ setHeloYOffset(rParam: UINT) : void	
+ getModeSpeed() : UINT	
+ getRcvrGain() : UINT	
+ getStab() : UINT	
+ getEraseGPI() : UINT	
+ getPersist() : UINT	
+ getRange() : UINT	
+ getHeloXPos() : UINT	
+ getHeloYPos() : UINT	
+ -RadarCtg() : void	



RADAR beam propagation model and calculations are independent from the rest of the software.



Network thread implementation receives data and executes separate from the rest of the software.



0.2

FIG 4 BEST AVAILABLE COPY

